Amendments to the Claims

This listing of claims will replace all prior versions, and listings of claims in the application.

Claims 1-22. (Currently cancelled).

Claim 23. (Amended) A superscalar microprocessor ~~system~~ capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor ~~system~~ comprising:

(a) an instruction fetch unit configured to ~~fetch instructions from an instruction store and configured to~~ provide a plurality of ~~the~~ instructions to an instruction buffer;

(b) an execution unit, coupled to the instruction fetch unit, configured to execute the plurality of ~~the~~ instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so ~~that a~~ the one load request can be made before a memory request, wherein the one load request ~~corresponding~~ corresponds to ~~an~~ a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction ~~that~~ precedes ~~a load instruction corresponding to the load request~~ the first instruction in the program order ~~and store requests in program order~~, the load store unit including:

(i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system;

(ii) load dependency ~~check~~ detection circuitry, wherein the load store unit does not make a particular load request when the load dependency ~~check~~ detection circuitry detects an address collision or write pending for that particular load request; and

(iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one ~~instruction in the one or more~~ of the plurality of instructions from the instruction buffer in a clock cycle.

Claim 24. (Amended) The ~~system~~ microprocessor according to claim 23, wherein the execution unit further comprises address generation circuitry adapted to generate addresses for the load and store requests, wherein an address for a load request may be generated out-of-order.

Claim 25. (Amended) The ~~system~~ microprocessor according to claim 23, wherein the execution unit further comprises address generation circuitry adapted to generate addresses for the load and store requests, wherein an address for a store request may be generated out-of-order.

Claim 26. (Amended) A superscalar microprocessor ~~system~~ capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor ~~system~~ comprising:

(a) an instruction fetch unit ~~for fetching instructions from an instruction store and for providing~~ configured to provide a plurality of ~~the~~ instructions to an instruction buffer;

(b) an execution unit, coupled to the instruction fetch unit, ~~for executing~~ configured to execute the plurality of the instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that ~~a~~ the one load request can be made before a memory request, wherein the one load request ~~corresponding~~ corresponds to ~~an~~ a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction ~~that~~ precedes ~~a load~~ the first instruction ~~corresponding to the load request~~ in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address and a store address may be generated out of the program order;

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system; and

(iii) a data path ~~for transferring~~ configured to transfer load data from the memory system to the execution unit,

wherein the superscalar microprocessor initiates execution of more than one ~~instruction in the one or more instructions~~ of the plurality of instructions from the instruction buffer in a clock cycle.

Claim 27. (Amended) The ~~system~~ microprocessor according to claim 26, ~~wherein the data path includes~~ further including alignment control circuitry ~~for generating~~ configured to generate a plurality of memory requests in response to a single instruction in the plurality of ~~the~~ instructions when an operand of the single instruction falls on a word boundary.

Claim 28. (Amended) The ~~system~~ microprocessor according to claim 27, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

Claim 29. (Amended) The ~~system~~ microprocessor according to claim 27, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

Claim 30. (Amended) The ~~system~~ microprocessor according to claim 26, wherein the load store unit comprises dependency ~~check~~ detection circuitry adapted to detect store-to-load dependencies, wherein the ~~store-to-load~~ dependency detection circuitry determines when data for a load request depends on a store request.

Claim 31. (Amended) The ~~system~~ microprocessor according to claim 30, wherein the dependency ~~check~~ detection circuitry includes address comparison logic ~~for comparing~~ configured to compare an address of a load request and an address of a store request.

Claim 32. (Amended) The ~~system~~ microprocessor according to claim 30, wherein the dependency ~~check~~ detection circuitry includes relative age determining logic ~~for determining~~ configured to determine the relative ~~age~~ program order of a load request corresponding to a first memory instruction in the plurality of instructions and a store request corresponding to a second memory instruction in the plurality of instructions.

Claim 33. (Amended) A computer system, comprising:

(a) a memory system ~~that is~~ configured to ~~store~~ retain instructions and data, the instructions having a program order;

(b) a superscalar processor~~, connected to the memory system, for executing~~ configured to execute the instructions, wherein the superscalar ~~micro~~processor ~~initiates~~ is configured to initiate more than one instruction in a clock cycle, the processor having,

(1) an instruction fetch unit ~~for fetching instructions from the memory system and for providing~~ configured to provide a plurality of ~~the~~ instructions to an instruction buffer;

(2) an execution unit, coupled to the instruction fetch unit, ~~for executing~~ configured to execute the plurality of ~~the~~ instructions from the instruction buffer in an out-of-order fashion, the execution unit including,

(i) a register file;

(ii) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order and

(iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that ~~a~~ the one load request can be made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction ~~corresponding to an instruction that~~ precedes ~~a load~~ the first instruction ~~corresponding to the load request~~ in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file.

Claim 34. (Amended) The system according to claim 33, wherein the address generation circuitry is further adapted to generate addresses for the load and store requests as soon as all operands are valid and the address generation circuitry is available for address generation~~execution unit further comprises address generation circuitry adapted to generate addresses for the load requests and store requests, wherein an address for a load request is generated out-of-order~~.

Claim 35. (Amended) The system according to claim 33, wherein the generated addresses include linear and physical addresses, and the address generation circuitry is further adapted to general physical addresses corresponding to linear addresses~~execution~~

~~unit further comprises address generation circuitry adapted to generate addresses for the load requests and store requests, wherein an address for a store request is generated out-of-order~~.

Claim 36. (Amended) The system according to claim 33, wherein the load_[[/]]store unit includes alignment control circuitry ~~for generating~~ configured to generate a plurality of memory requests in response to a single instruction in the plurality of ~~the~~ instructions when an operand of the single instruction falls on a word boundary.

Claim 37. (Previously presented) The system according to claim 36, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

Claim 38. (Previously presented) The system according to claim 36, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

Claim 39. (Amended) The system according to claim 33, wherein the load store unit comprises dependency ~~check~~ detection circuitry adapted to detect store-to-load dependencies, wherein the ~~store-to-load~~ dependency detection circuitry determines when data for a load request depends on a store request.

Claim 40. (Amended) The system according to claim 39, wherein the dependency ~~check~~ detection circuitry includes address comparison logic ~~for comparing~~ configured to compare an address of a load request and an address of a store request.

Claim 41. (Amended) The system according to claim 39, wherein the dependency ~~check~~ detection circuitry includes relative age determining logic ~~for determining~~ configured to determine the relative ~~age~~ program order of a load request corresponding to a first memory instruction in the plurality of instructions and a store request corresponding to a second memory instruction in the plurality of instructions.

Claim 42. (Amended) A superscalar microprocessor ~~system~~ capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor ~~system~~ comprising:

(a) an instruction fetch unit ~~for fetching instructions from an instruction store and for providing~~ configured to provide a plurality of the instructions to an instruction buffer;

(b) an execution unit, coupled to the instruction fetch unit, ~~for executing~~ configured to execute the plurality of ~~the~~ instructions from the instruction buffer in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that ~~a~~ the one load request can be made before a memory request, wherein the one load request corresponds ~~corresponding~~ to ~~an~~ a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second

instruction ~~that~~ precedes ~~a load~~ the first instruction ~~corresponding to the load request~~ in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses for instructions in the instruction buffer, wherein at least one of a load address ~~is~~ and a store address may be generated out of the program order;

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system; ~~and~~

(iii) dependency ~~check~~ detection circuitry adapted to detect store-to-load dependencies, wherein the ~~store-to-load~~ dependency detection circuitry determines when data for a load request depends on a store request; and

(iv) a data path ~~for transferring~~ configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one ~~instruction in the one or more~~ of the plurality of instructions from the instruction buffer in a clock cycle.

Claim 43. (Amended) The ~~system~~ microprocessor according to claim 42, ~~wherein the data path includes~~ further including alignment control circuitry ~~for generating~~ configured to generate a plurality of memory requests in response to a single

instruction in the plurality of ~~the~~ instructions when an operand of the single instruction falls on a word boundary.

Claim 44. (Amended) The ~~system~~ microprocessor according to claim 43, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

Claim 45. (Amended) The ~~system~~ microprocessor according to claim 43, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

Claim 46. (Amended) The ~~system~~ microprocessor according to claim 42, wherein the dependency ~~check~~ detection circuitry includes relative age determining logic ~~for determining~~ configured to determine the relative ~~age~~ program order of a load instruction in the plurality of the instructions and a store instruction in the plurality of the instructions.

Claim 47. (New) The microprocessor according to claim 23, wherein the load store unit is further adapted to make store requests in the program order.

Claim 48. (New) The microprocessor according to claim 23, wherein the execution unit further comprises address generation circuitry adapted to generate

addresses for the load and store requests when all operands are valid and the address generation circuitry is available for address generation.

Claim 49. (New) The microprocessor according to claim 23, wherein the execution unit further comprises address generation circuitry adapted to generate linear addresses for the load and store requests, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a scaled index register.

Claim 50. (New) The microprocessor according to claim 23, wherein the execution unit further comprises address generation circuitry adapted to generate addresses for the load and store requests, including generation of linear addresses and corresponding physical addresses.

Claim 51. (New) The microprocessor according to claim 23, wherein the load store unit is further adapted to make memory-mapped input/output (I/O) requests according to the program order.

Claim 52. (New) The microprocessor according to claim 23, wherein the data path is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 53. (New) The microprocessor according to claim 26, wherein the load store unit is further adapted to make store requests in the program order.

Claim 54. (New) The microprocessor according to claim 26, wherein the address generation unit is further configured to generate load and store addresses when all operands are valid and the address generation unit is available for address generation.

Claim 55. (New) The microprocessor according to claim 26, wherein the generated load and store addresses include linear and physical addresses, and the address generation unit is further configured to generate physical addresses corresponding to linear addresses.

Claim 56. (New) The microprocessor according to claim 26, wherein the load store unit is adapted to make memory-mapped input/output (I/O) requests according to the program order.

Claim 57. (New) The microprocessor according to claim 26, wherein the data path is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 58. (New) The system according to claim 33, wherein the load store unit is further adapted to make store requests in the program order.

Claim 59. (New) The system according to claim 33, wherein the load store unit is further adapted to make memory-mapped input/output (I/O) load requests in the program order.

Claim 60. (New) The system according to claim 33, wherein the load store unit is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 61. (New) The microprocessor according to claim 42, wherein the load store unit is further adapted to make store requests in the program order.

Claim 62. (New) The microprocessor according to claim 42, wherein the address generation unit is further configured to generate load and store addresses as soon as all operands are valid and the address generation unit is available for address generation.

Claim 63. (New) The microprocessor according to claim 42, wherein the address generation unit is further configured to generate linear load and store addresses, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a scaled index register.

Claim 64. (New) The microprocessor according to claim 42, wherein the address generation unit is further configured to generate linear load and store addresses, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a displacement.

Claim 65. (New) The microprocessor according to claim 42, wherein the generated load and store addresses include linear and physical addresses, and the address generation unit is further configured to generate physical addresses corresponding to linear addresses.

Claim 66. (New) The microprocessor according to claim 42, wherein the load store unit is further adapted to make memory-mapped input/output (I/O) load requests in the program order.

Claim 67. (New) The microprocessor according to claim 42, wherein the data path is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 68. (New) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load

requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so the one load request can be made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit including:

(i) an address path adapted to manage load and store addresses and to provide the load and store addresses to the memory system;

(ii) load dependency detection circuitry, wherein the load store unit does not make a particular load request when the load dependency detection circuitry detects an address collision or write pending for that particular load request; and

(iii) a data path adapted to transfer data from the memory system to the execution unit in response to load requests, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle.

Claim 69. (New) The microprocessor according to claim 68, wherein the execution unit further comprises address generation circuitry adapted to generate addresses for the load and store requests, wherein an address for a load request may be generated out-of-order.

Claim 70. (New) The microprocessor according to claim 68, wherein the execution unit further comprises address generation circuitry adapted to generate addresses for the load and store requests, wherein an address for a store request may be generated out-of-order.

Claim 71. (New) The microprocessor according to claim 68, wherein the load store unit is further adapted to make store requests in the program order.

Claim 72. (New) The microprocessor according to claim 68, wherein the execution unit further comprises address generation circuitry adapted to generate addresses for the load and store requests when all operands are valid and the address generation circuitry is available for address generation.

Claim 73. (New) The microprocessor according to claim 68, wherein the execution unit further comprises address generation circuitry adapted to generate linear addresses for the load and store requests, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a scaled index register.

Claim 74. (New) The microprocessor according to claim 68, wherein the execution unit further comprises address generation circuitry adapted to generate addresses for the load and store requests, including generation of linear addresses and corresponding physical addresses.

Claim 75. (New) The microprocessor according to claim 68, wherein the load store unit is further adapted to make memory-mapped input/output (I/O) load requests according to the program order.

Claim 76. (New) The microprocessor according to claim 68, wherein the data path is further adapted to merge data returning from the memory system with initial contents of a destination register.

Claim 77. (New) The microprocessor according to claim 68, wherein the execution unit is further configured to merge data returning from the memory system with initial contents of a destination register.

Claim 78. (New) The microprocessor according to claim 68, further comprising an instruction fetch unit configured to provide the plurality of instructions to an instruction buffer, wherein the execution unit executes the plurality of instructions from the instruction buffer in an out of order fashion.

Claim 79. (New) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load

requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request can be made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system; and

(iii) a data path configured to transfer load data from the memory system to the execution unit,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle.

80. (New) The microprocessor according to claim 79, further including alignment control circuitry configured to generate a plurality of memory requests in response to a single instruction in the plurality of instructions when an operand of the single instruction falls on a word boundary.

81. (New) The microprocessor according to claim 80, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

82. (New) The microprocessor according to claim 80, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

83. (New) The microprocessor according to claim 79, wherein the load store unit comprises dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request.

84. (New) The microprocessor according to claim 83, wherein the dependency detection circuitry includes address comparison logic configured to compare an address of a load request and an address of a store request.

85. (New) The microprocessor according to claim 83, wherein the dependency detection circuitry includes relative age determining logic configured to determine the relative program order of a load request corresponding to a first memory instruction in the plurality of instructions and a store request corresponding to a second memory instruction in the plurality of instructions.

86. (New) The microprocessor according to claim 79, wherein the load store unit is further adapted to make store requests in the program order.

87. (New) The microprocessor according to claim 79, wherein the address generation unit is further configured to generate load and store addresses when all operands are valid and the address generation unit is available for address generation.

88. (New) The microprocessor according to claim 79, wherein the generated load and store addresses include linear and physical addresses, and the address generation unit is further configured to generate physical addresses corresponding to linear addresses.

89. (New) The microprocessor according to claim 79, wherein the load store unit is adapted to make memory-mapped input/output (I/O) load requests according to the program order.

90. (New) The microprocessor according to claim 79, wherein the execution unit is further configured to merge data returning from the memory system with initial contents of a destination register.

91. (New) The microprocessor according to claim 79, wherein the data path is further configured to merge data returning from the memory system with initial contents of a destination register.

92. (New) The microprocessor according to claim 79, further comprising an instruction fetch unit configured to provide the plurality of instructions to an instruction

buffer, wherein the execution unit executes the plurality of instructions from the instruction buffer in an out of order fashion.

93. (New) A superscalar microprocessor configured to initiate execution of more than one instruction in a clock cycle, the processor comprising:

(a) a memory system configured to retain instructions and data, the instructions having a program order;

(b) an execution unit configured to execute the plurality of instructions in an out-of-order fashion, the execution unit including,

(iii) a register file;

(iv) address generation circuitry adapted to generate addresses for load requests and store requests out-of-order; and

(iii) a load store unit adapted to make the load requests and the store requests to the memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request can be made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, wherein the second instruction precedes the first instruction in the program order, the load store unit further adapted to return data falling on a word boundary in correct alignment to the register file.

94. (New) The microprocessor according to claim 93, wherein the address generation circuitry is further adapted to generate addresses for the load and store

requests when all operands are valid and the address generation circuitry is available for address generation.

95. (New) The microprocessor according to claim 93, wherein the generated addresses include linear and physical addresses, and the address circuitry is further adapted to generate physical addresses corresponding to linear addresses.

96. (New) The microprocessor according to claim 93, wherein the load store unit includes alignment control circuitry configured to generate a plurality of memory requests in response to a single instruction in the plurality of instructions when an operand of the single instruction falls on a word boundary.

97. (New) The microprocessor according to claim 96, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

98. (New) The microprocessor according to claim 96, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

99. (New) The microprocessor according to claim 93, wherein the load store unit comprises dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request.

100. (New) The microprocessor according to claim 99, wherein the dependency detection circuitry includes address comparison logic configured to compare an address of a load request and an address of a store request.

101. (New) The microprocessor according to claim 99, wherein the dependency detection circuitry includes relative age determining logic configured to determine the relative program order of a load request corresponding to a first memory instruction in the plurality of instructions and a store request corresponding to a second memory instruction in the plurality of instructions.

102. (New) The microprocessor according to claim 93, wherein the load store unit is further adapted to make store requests in the program order.

103. (New) The microprocessor according to claim 93, wherein the load store unit is further adapted to make memory-mapped input output (I/O) load requests in the program order.

104. (New) The microprocessor according to claim 93, wherein the load store unit is further adapted to merge data returning from the memory system with initial contents of a destination register.

105. (New) The microprocessor according to claim 93, wherein the execution unit further includes merge data circuitry configured to merge data returning from the memory system with initial contents of a destination register.

106. (New) The microprocessor according to claim 93, further comprising an instruction fetch unit configured to provide the plurality of instructions to an instruction buffer, wherein the execution unit executes the plurality of instructions from the instruction buffer in an out of order fashion.

107. (New) A superscalar microprocessor capable of executing one or more instructions out-of-order with respect to an ordering defined by a program order, the microprocessor comprising:

an execution unit configured to execute a plurality of instructions in an out-of-order fashion, the execution unit including a load store unit adapted to make load requests and store requests to a memory system, the load store unit adapted to make at least one load request out of the program order so that the one load request can be made before a memory request, wherein the one load request corresponds to a first instruction from the plurality of instructions and the memory request corresponds to a second instruction from the plurality of instructions, and wherein the second instruction precedes the first instruction in the program order, the load store unit having,

(i) an address generation unit configured to generate load and store addresses out of order for instructions in the plurality of instructions;

(ii) an address path adapted to manage the generated load and store addresses and to provide the generated load and store addresses to the memory system;

(iii) dependency detection circuitry adapted to detect store-to-load dependencies, wherein the dependency detection circuitry determines when data for a load request depends on a store request; and

(iv) a data path configured to transfer load data from the memory system to the execution unit, the data path configured to align data returned from the memory system to thereby permit data falling on a word boundary to be returned from the memory system to the execution unit in correct alignment,

wherein the superscalar microprocessor initiates execution of more than one of the plurality of instructions in a clock cycle.

108. (New) The microprocessor according to claim 107, further including alignment control circuitry configured to generate a plurality of memory requests in response to a single instruction in the plurality of instructions when an operand of the single instruction falls on a word boundary.

109. (New) The microprocessor according to claim 108, wherein the single instruction is a load instruction and the plurality of memory requests are load requests.

110. (New) The microprocessor according to claim 108, wherein the single instruction is a store instruction and the plurality of memory requests are store requests.

111. (New) The microprocessor according to claim 107, wherein the dependency detection circuitry includes relative age determining logic configured to determine the relative program order of a load instruction in the plurality of the instructions and a store instruction in the plurality of the instructions.

112. (New) The microprocessor according to claim 107, wherein the load store unit is further adapted to make store requests in the program order.

113. (New) The microprocessor according to claim 107, wherein the address generation unit is further configured to generate load and store addresses as soon as all operands are valid and the address generation unit is available for address generation.

114. (New) The microprocessor according to claim 107, wherein the address generation unit is further configured to generate linear load and store addresses, the linear address generation including the addition of three or more address components, the address components including a segment base, a base register, and a scaled index register.

115. (New) The microprocessor according to claim 107, wherein the generated load and store addresses include linear and physical addresses, and the address generation unit is further configured to generate physical addresses corresponding to linear addresses.

116. (New) The microprocessor according to claim 107, wherein the load store unit is further adapted to make memory-mapped input/output (I/O) load requests in the program order.

117. (New) The microprocessor according to claim 107, wherein the data path is further configured to merge data returning from the memory system with initial contents of a destination register.

118. (New) The microprocessor according to claim 107, wherein the load store unit includes merge data circuitry configured to merge data returning from the memory system with initial contents of a destination register.

119. (New) The microprocessor according to claim 107, wherein the execution unit is further configured to merge data returning from the memory system with initial contents of a destination register.

120. (New) The microprocessor according to claim 107, wherein the execution unit is further configured to provide store data to the data path as load data when the dependency detection circuitry determines that data for a load request depends on a store request.

121. (New) The microprocessor according to claim 120, wherein the execution unit is further configured to provide data stored by a store request as load data by way of the memory system.

122. (New) The microprocessor according to claim 107, further comprising an instruction fetch unit configured to provide the plurality of instructions to an instruction buffer, wherein the execution unit executes the plurality of instructions from the instruction buffer in an out of order fashion.